

Подпрограммы на языке Pascal

Подпрограммы используют при создании больших программ, в которых содержатся повторяющиеся группы операторов.

Повторяющиеся операторы оформляют в виде самостоятельной программы и называют её подпрограммой. Каждая подпрограмма имеет имя, дающее возможность к ней обращаться в соответствующих местах программы.

Подпрограмма – это программа внутри большой головной программы, имеет такую же структуру, что и программа

- Информация между основной программой и подпрограммой передаётся **глобальными параметрами**, действующими в любой части программы. Глобальные параметры описываются в основной программе.
- Внутри подпрограммы используют **локальные параметры** – их имена и значения описываются и действуют в пределах данной подпрограммы и недоступны вызывающей программе.
- Локальные переменные являются формальными, они лишь резервируют место для фактических. При вызове процедуры формальные параметры заменяются фактическими



-
- В Паскале имеется два вида подпрограмм:
процедуры и функции.
 - Их структура похожа на структуру основной программы.



□ Описание процедуры

PROCEDURE <имя> [(Список формальных параметров)];

[Описательная часть]

Label ;

Const;

Type;

Var;

Begin

Тело процедуры

End;

Количество и типы формальных и фактических параметров должны совпадать. Фактические параметры указываются в скобках при вызове процедуры. Они заменяют формальные параметры при выполнении программы



В основной программе процедура вызывается
тоже с указанием параметров:

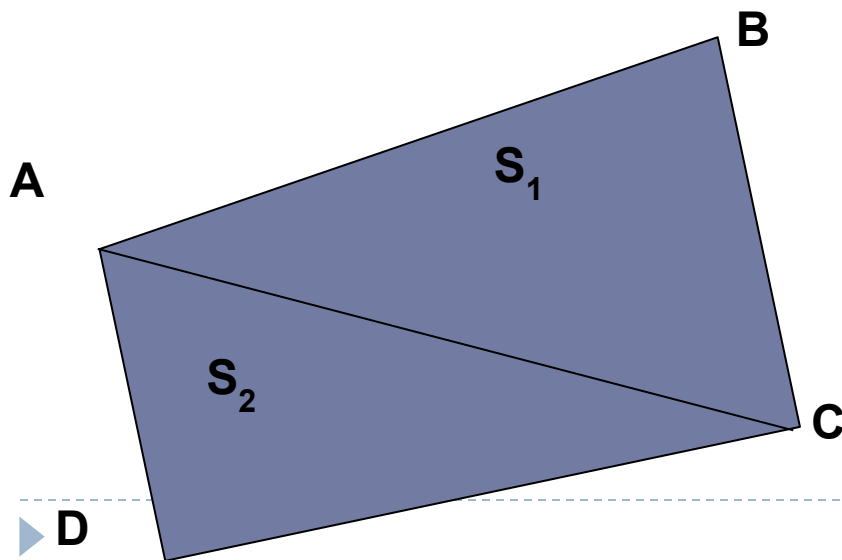
<ИМЯ> (Список параметров);

Например,

Ploshad3 (AB, BC, AC, S);

Пример:

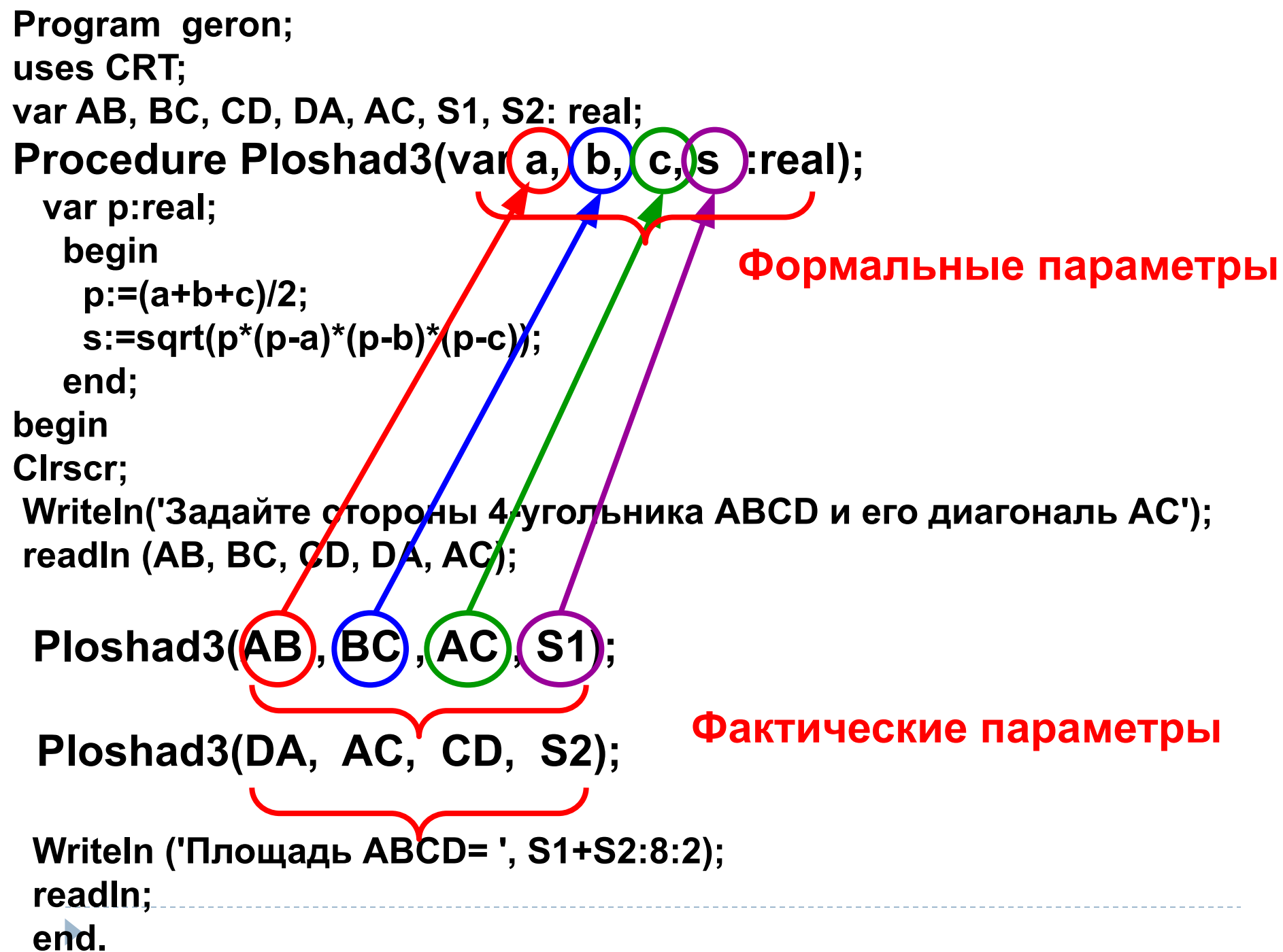
Вычислить площадь выпуклого четырёхугольника, заданного
длинами 4-х сторон и диагональю



*По формуле Герона можно вычислить
площадь треугольника*

$$S = \sqrt{p \cdot (p - a) \cdot (p - b) \cdot (p - c)}$$

$$p = \frac{a + b + c}{2}$$



□ Описание функции

Function <имя> [(Список формальных параметров)]: тип;

[Описательная часть]

Label ;

Const;

Type;

Var;

Begin

Тело функции, в которой обязательно должно быть присваивание

Имя_ функции_ := значение

End;

Обращение к функции производится по имени функции с указанием фактических параметров



Составить программу, подсчитывающую число сочетаний без повторений из n элементов по k

Известно, что число сочетаний из n элементов по k вычисляется по формуле

$$C = \frac{n!}{k!(n-k)!}$$

Алгоритм вычисления факториала

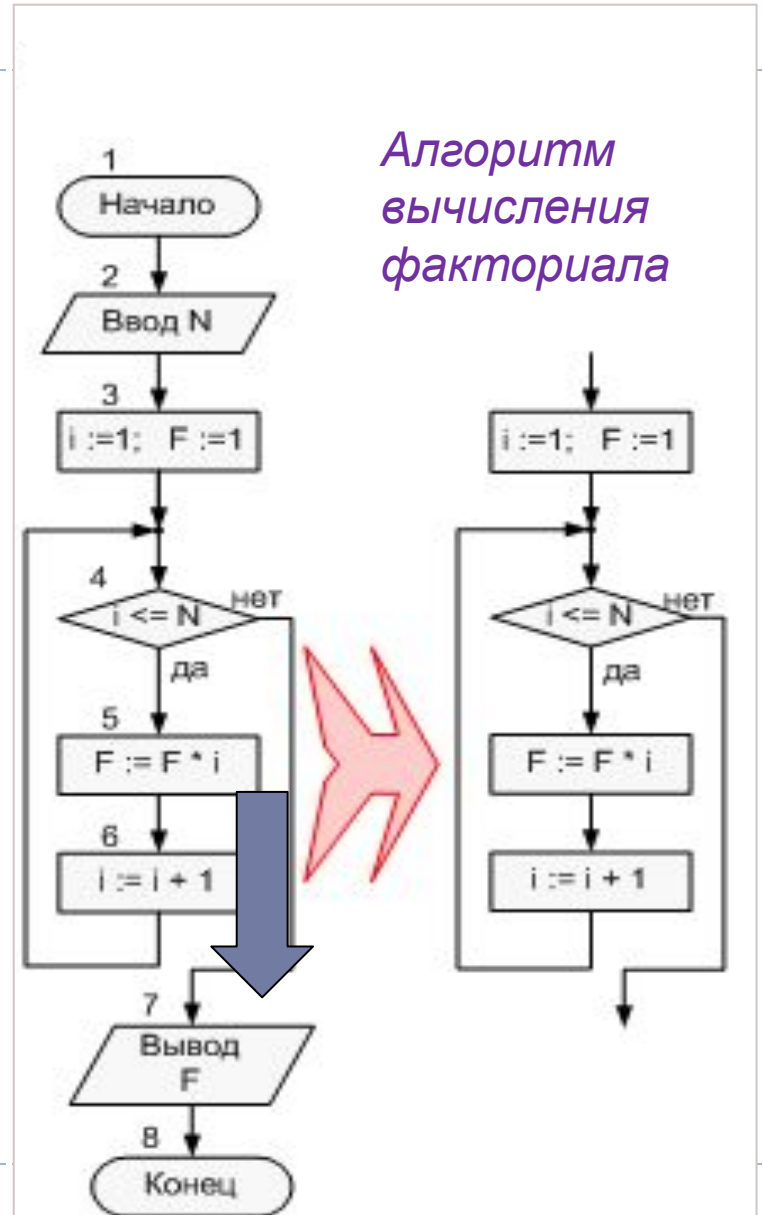


Схема
головного алгоритма

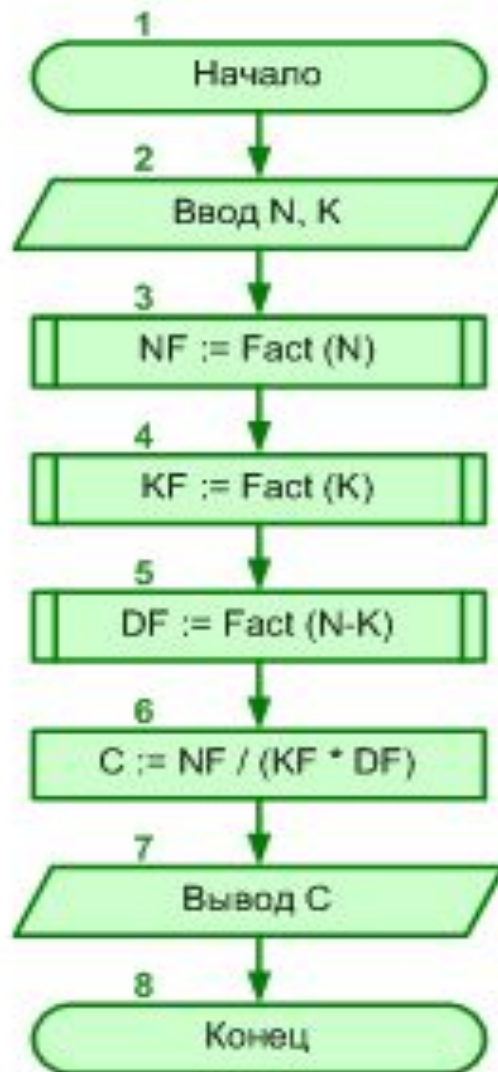
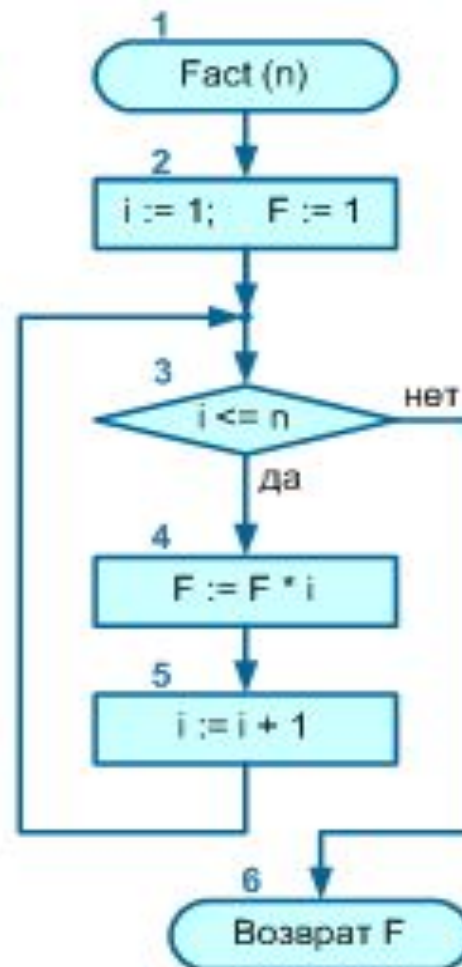


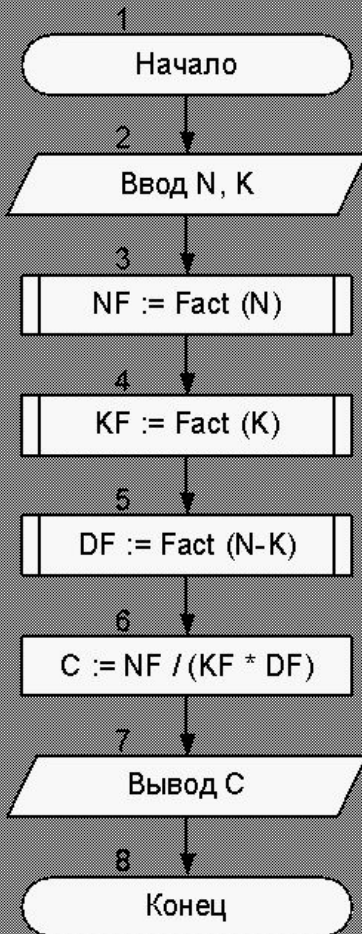
Схема
подалгоритма
вычисления
факториала



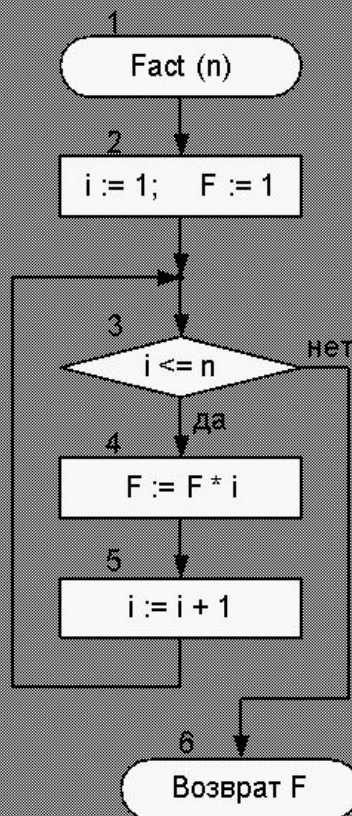
Задача 1. Составить программу, определяющую число сочетаний из N по K . Выполнить её для $N=4, K=2$.

2 Схемы составного алгоритма:

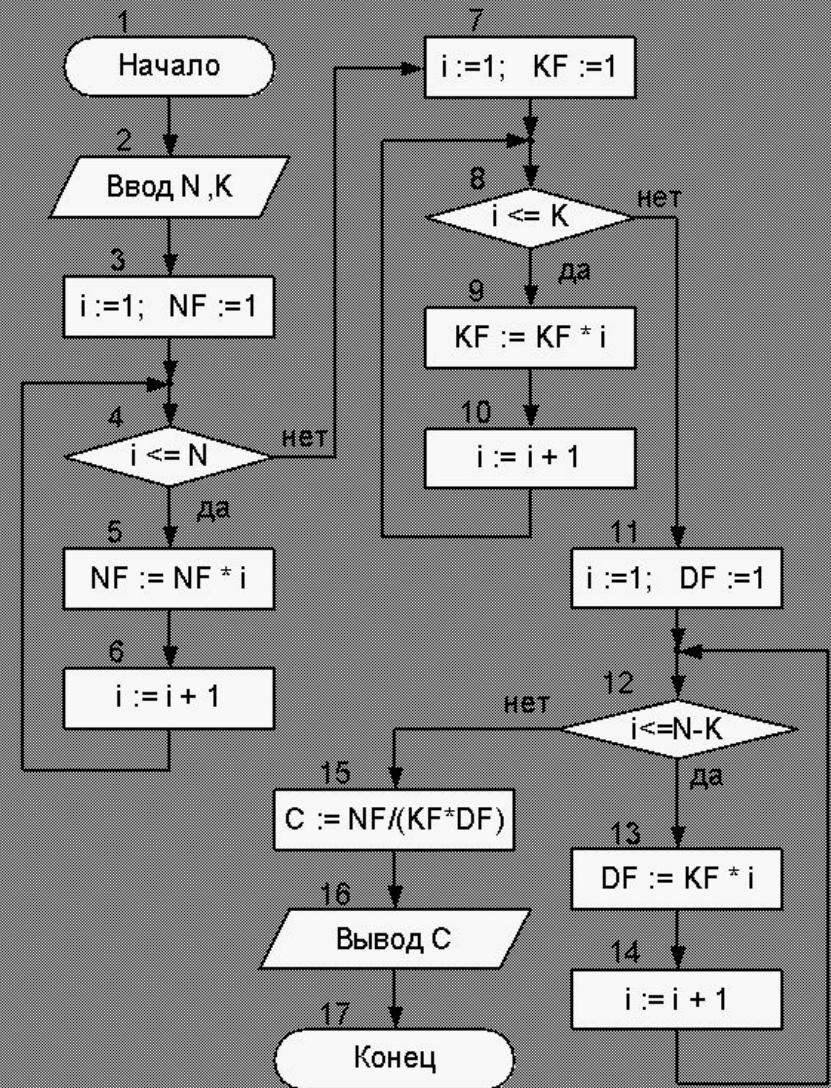
Головной алгоритм



Подалгоритм
вычисления
факториала



1 Схема «простого» алгоритма,
решающего эту же задачу без подпрограмм



```
Program sochetaniya;  
Uses CRT;  
Var n,k:integer;nf,kf,df:longint;c:real;  
Function Fact(n:integer):longint;  
    var i:integer;F:longint;  
    begin  
        F:=1;  
        while i<=n do begin  
            F:=F*i ;  
            i:=i+1;  
            Fact:=F;  
        end;  
    end;  
  
Begin  
write('n=');  
readln(n);  
    write('k=');  
readln(k);  
nf:=fact(n);  
kf:=fact(k);  
df:=fact(n-k);  
c:=nf/kf*df;  
writeln('C(n,k)=' ,c);  
end.
```